AF

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on January 27, 2006.

Attorney for Applicant(s)

PATENT APPLICATION
Docket No. AUZ-001 P

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s):	David M. Austin et al.	
Serial No.:	09/491,727	
Filed:	January 27, 2000	Group Art Unit: 2131
For:	DETECTION OF OBSERVER PROGRAMS AND COUNTERMEASURES AGAINST OBSERVER PROGRAMS	
Examiner:	Syed Zia	

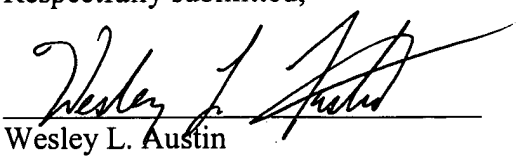
REPLY TO NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

A second Notification of Non-Compliant Appeal Brief was mailed on December 28, 2005. Appellants' hereby reply to this Notification by filing a corrected Appeal Brief.

Respectfully submitted,



Wesley L. Austin
Reg. No. 42,273
Attorney for Appellant(s)

Date: January 27, 2006

Wesley L. Austin, Esq.
Trapware Corporation
1244 E. 1650 S.
Bountiful, UT 84010
Telephone: (801) 537-1700

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on January 27, 2006.

Attorney for Applicant(s)

PATENT APPLICATION
Docket No. AUZ-001 P

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s):	David M. Austin et al.	
Serial No.:	09/491,727	
Filed:	January 27, 2000	Group Art Unit: 2131
For:	DETECTION OF OBSERVER PROGRAMS AND COUNTERMEASURES AGAINST OBSERVER PROGRAMS	
Examiner:	Syed Zia	

APPELLANTS' APPEAL BRIEF – TWICE CORRECTED

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

An Office Action dated April 21, 2005 rejected all claims (claims 1-32) in the present application. A timely Notice of Appeal was mailed on June 6, 2005 and was received by the United States Patent Office on June 9, 2005. Appellants' Appeal Brief is being filed herewith. This Appeal Brief is being filed in triplicate under the provisions of 37 C.F.R. § 1.192.

1. REAL PARTY IN INTEREST

The real party in interest is the assignee, Trapware Corporation.

2. RELATED APPEALS AND INTERFERENCES

There are no related appeals and interferences.

3. STATUS OF CLAIMS

Claims 1-32 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Drake, U.S. Patent No. 6,006,328 (hereinafter, "Drake").

Appellants appeal the rejections of claims 1-32.

4. STATUS OF AMENDMENTS

No amendments were filed subsequent to the rejection.

5. SUMMARY OF INVENTION

As stated in the background section of the patent application, software has been developed to observe or monitor computer users. These software programs provide a wide variety of monitoring features. For example, some of these programs are able to log keystrokes of a user, log menu commands, take screen shots of a user's computer screen at various times, track use of various programs, track what web sites have been visited, monitor e-mail communications, etc. With the technology available today, most, if not all, of a computer user's activities on a computer can be observed and recorded. See the Appellants' patent application (hereinafter referred to as the "Specification"), page 3, lines 1-22.

With the computer technology of today and with the observing programs now available and for those programs that will surely be developed and used in the future, computer users may be watched by third parties more often than many think. It would be highly beneficial to computer users if they could find out whether they are being observed by computer software and technology and to know information about the observing activity and/or program. Specification, page 3, lines 1-22.

As presently claimed, a new system has been developed for detecting an observing program on a computer system as including accessing instructions that access observer data. One or more embodiments of an observer program are described in the Specification on page 14, lines 9-23, page

15, lines 1-14, and Figure 2. The observer data includes data descriptive of the observer program. The observer program is programmed to observe a user's activities on the computer system and also operates to create data from its observations. The system also includes reading instructions that read memory of the computer system to obtain memory data. Further, the system includes comparing instructions that compare the observer data with memory data read in from memory to determine whether the observer program is present on the computer system. One or more embodiments of the system and how it detects an observer program are described in the Specification on page 17, lines 8-22, page 18, lines 1-23, page 19, lines 1-23, page 20, lines 1-23, page 21, lines 1-23, 9-23, and Figure 3. The system also includes generating instructions that generate results from the reading and comparing. The results generated indicate whether the observer program is present on the computer system. In addition, the system includes outputting instructions that obtain the results and provide the results for a user. The outputting instructions may provide the results to a user through a graphical user interface.

As required by 37 C.F.R. § 41.37(c)(1)(v), a summary of claimed subject matter immediately follows. The references to the specification refer only to embodiments of the invention. The invention is defined by the claims. Accordingly, these references to the specification are not meant to limit the scope of the claims of the present invention in any way but are only provided because they are mandated by 37 C.F.R. § 41.37(c)(1)(v). All references are to the patent specification.

1. A system for detecting the presence of an observing program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the system including computer software for running on the computer system, the system comprising:

observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program; (pg. 9, lines 6-8; pg.

10, lines 4-15; pg. 14, lines 9-23; pg. 15, lines 1-23; pg. 16, lines 1-23; pg. 17, lines 1-7; Figure 2, elements 34-48; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23)

accessing instructions that access the observer data; (pg. 9, lines 5-6; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

reading instructions that read memory of the computer system to obtain memory data; (pg. 9, lines 8-10; pg. 10, lines 16-23; pg. 11, lines 1-8; pg, 11, lines 9-18; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

comparing instructions that compare the observer data with memory data read in from memory to determine whether the observer program is present on the computer system; (pg. 9, lines 10-12, 15-24; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

generating instructions that generate results from the comparing, wherein the results generated indicate whether the observer program is present on the computer system; and (pg. 9, lines 12-15, 15-24; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 22, lines 5-11; Figure 4; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

outputting instructions that obtain the results and provide the results for a user. (pg. 10, lines 1-3; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 22, lines 5-11; Figure 4; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

16. A system for detecting the presence of an observing program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the system including computer software for running on the computer system, the system comprising:

observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program; (pg. 9, lines 6-8; pg. 10, lines 4-15; pg. 14, lines 9-23; pg. 15, lines 1-23; pg. 16, lines 1-23; pg. 17, lines 1-7; Figure 2, elements 34-48; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23)

means for accessing the observer data; (pg. 9, lines 5-6; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

means for reading memory of the computer system to obtain memory data; (pg. 9, lines 8-10; pg. 10, lines 16-23; pg. 11, lines 1-8; pg, 11, lines 9-18; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

means for comparing the observer data with memory data to determine whether the observer program is present on the computer system; (pg. 9, lines 10-12, 15-24; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

means for generating results from the comparison, wherein the results generated indicate whether the observer program is present on the computer system; and (pg. 9, lines

12-15, 15-24; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 22, lines 5-11; Figure 4; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

means for outputting the results for a user. (pg. 10, lines 1-3; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 22, lines 5-11; Figure 4; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

17. A method for detecting the presence of an observing program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the method being implemented through computer software for running on the computer system, the method comprising the steps of:

accessing observer data, the observer data including data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program; ; (pg. 9, lines 5-6; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

reading memory of the computer system to obtain memory data; (pg. 9, lines 8-10; pg. 10, lines 16-23; pg. 11, lines 1-8; pg, 11, lines 9-18; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

comparing the observer data with memory data read in from memory to determine whether the observer program is present on the computer system; (pg. 9, lines 10-12, 15-24; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 18, lines 1-23; pg. 19, lines 1-

23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

generating results from the reading and comparing, wherein the results generated indicate whether the observer program is present on the computer system; and (pg. 9, lines 12-15, 15-24; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 22, lines 5-11; Figure 4; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

outputting the results for a user. (pg. 10, lines 1-3; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 22, lines 5-11; Figure 4; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

20. A system for altering the operation of an observer program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the system including computer software for running on the computer system, the system comprising:

observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program; (pg. 9, lines 6-8; pg. 10, lines 4-15; pg. 14, lines 9-23; pg. 15, lines 1-23; pg. 16, lines 1-23; pg. 17, lines 1-7; Figure 2, elements 34-48; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23)

accessing instructions that access the observer data; (pg. 9, lines 5-6; pg. 10, lines 16-23; pg. 11, lines 1-8; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24,

lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

reading instructions that read memory of the computer system to obtain files relating to the observer program; (pg. 9, lines 8-10; pg. 10, lines 16-23; pg. 11, lines 1-8; pg, 11, lines 9-18; pg. 17, lines 8-23; pg. 18, lines 1-23; pg. 19, lines 1-23; pg. 20, lines 1-23; pg. 21, lines 1-23; Figure 3, elements 50-60; pg. 23, lines 5-23; pg. 24, lines 1-23; pg. 25, lines 1-23; pg. 26, lines 1-23; pg. 27, lines 1-23; Figure 5; Figure 6)

altering instructions that alter a file relating to the observer program such that the operation of the observer program is changed. (pg. 28, lines 11-23; pg. 29, lines 1-23; pg. 30, lines 1-23; Figure 7)

6. ISSUES

The following issues are presented for review:

I. Whether claims 1-32 are unpatentable under 35 U.S.C. § 102(e) as being anticipated by Drake.

7. GROUPING OF CLAIMS

Claims 1-32 stand or fall together.

8. ARGUMENT

Claims 1-32 Rejected under 35 U.S.C. § 102

Claims 1-15

The Examiner rejected claims 1-32 under 35 U.S.C. § 102(e) as being anticipated by Drake, U.S. Patent No. 6,006,328 (hereinafter, "Drake"). This rejection is respectfully traversed.

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." M.P.E.P. § 2131 (July 1998) (citing Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ2d

1051, 1053 (Fed. Cir. 1987)). "The identical invention must be shown in as complete detail as is contained in the . . . claim." M.P.E.P. § 2131 (July 1998) (citing Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989)). In addition, "the reference must be enabling and describe the applicant's claimed invention sufficiently to have placed it in possession of a person of ordinary skill in the field of the invention." In re Paulsen, 31 USPQ2d 1671, 1673 (Fed. Cir. 1994).

Claim 1 recites "accessing instructions that access observer data", and "observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create a log file from the observing of the observer program." Drake does not disclose these claim elements. The Examiner has cited the following portion of Drake as disclosing these claim elements:

> This invention seeks to provide computer software having enhanced security features, to a process which substantially enhances the security of computer software (hereafter referred to as the improved process) and to a method by which to apply said improved process (hereafter referred to as the applicator).
>
> The improved process consists of including computer code to automatically detect tampering of said computer software, and computer code to prevent the theft of ID-Data by replacing existing vulnerable (to rogue software eavesdropping or attack) software or operating system code with secure equivalents which utilise anti-spy techniques (as described later in this document).
>
> Preferably, the improved process also consists of including computer code to prevent decompilation, reverse-engineering, and disassembly by the inclusion of obfuscating code inserts, and the use of executable encryption.
>
> Preferably, the improved process also consists of including code to prevent execution-tracing and debugging by the use of code designed to detect and prevent these operations.
>
> Preferably, the improved process consists of, or also includes, human-recognisable audio-visual components which permit the authenticity of said computer software to be easily verified by the user on each invocation using techniques described later in this document.
>
> The idea which lead to the creation of this invention can be summarised as follows: If a piece of computer software that is executing can be shown to be the genuine article, and this software can protect itself against eavesdropping, and this software can prevent tampering of itself, then is it possible for this software to function in a secure manner, even within an insecure operating system. This invention permits the creation of such a piece of computer software--having a tangible, useful security advantage and hence improving its value.

Drake, Col. 3, lines 32-67.

This portion of Drake does not disclose "accessing instructions that access observer data", and "observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create a log file from the observing of the observer program." It does mention "rogue software eavesdropping" (Col. 3, lines 41-42) and "anti-spy techniques" (Col. 3, lines 43), but these generic terms do not disclose these claim elements. Claim 1 specifically requires "accessing instructions that access observer data," and "observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create a log file from the observing of the observer program."

Claim 1 also recites "comparing instructions that compare the observer data with memory data read in from memory to determine whether the observer program is present on the computer system." Drake does not disclose this claim element. The Examiner has cited the following portion of Drake as disclosing this claim element:

Aspect 3. Detecting Tampering

> As hereinbefore described, it is desirable to detect tampering, since this may lead to the reduction of software security.
>
> ***This can be achieved with the use of code which is protected from disassembly and examination through obfuscation and encryption, which re-reads its own external-image and compares it with its known memory image or precalculated check-data to detect hot-patching (ie. the modification of software sometime after it has been loaded from disk, but (usually) before execution of the modified section has commenced).***
>
> ***Additionally, the software can scan the memory image of itself one or more times, or continuously, to ensure that unexpected alterations do not occur.***
>
> Certain modifications to the external copy of software are reflected in subtle changes to the environment in which the modified software will be executed (for example: the size of the code, if altered, will be reflected in the initial code size value supplied to the executing program being incorrect.). Additionally, certain modification to the operating system and environment of said software can also be monitored (for example: certain interrupt vector table pointers in Intel-processor applications) to detect unexpected changes by rogue software. These changes can also be detected to prevent tampering.

> Once tampering is detected, program flow-of-control needs to be changed so that the potential compromise associated with ID-Data theft is avoided. This may be the security-enhanced program terminating with a message indicating that its integrity has been compromised before all of the ID Data is entered. Alternatively, the fact that tampering has been detected may be kept secret and the ID-Data retrieved, however, immediately upon retrieval, the ID-Data entered can be invalidated thus preventing access to that which the now potentially compromised ID-Data would have otherwise allowed. This latter method allows for the possibility of security-enhanced software informing remote or other authorities that tampering was detected and possibly other information, such as what specifically was altered and by whom. Care must be taken to ensure the integrity of the "remote-informing" code before ID-Data entry is permitted.

Drake, Col. 6, lines 5-48 (emphasis added).

This portion of Drake does not disclose "comparing instructions that compare the observer data with memory data read in from memory to determine whether the observer program is present on the computer system." This section of Drake discloses aspects of detecting tampering, as the heading in Drake indicates. Recall that claim 1 recited above "observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create a log file from the observing of the observer program."

Claim 1 further recites "outputting instructions that obtain the results and provide the results for a user." Claim 1 also states "wherein the results generated indicate whether the observer program is present on the computer system." Drake does not disclose this claim limitation. The Examiner has cited Col. 6, lines 5-48 (quoted above) and the following portion of Drake as disclosing this claim element:

> Detailed hereafter are several security-enhancing techniques to combat eavesdropping. Security is provided by (a) hampering examination of software-code operating system code or or parts thereof through the use of the encryption or partial encryption of said code, (b) preventing the disassembly of said code through the inclusion of dummy instructions and prefixes and additional code to mislead and hamper disassembly (ie: obfuscating inserts), (c) preventing the computerised tracing of the execution of said code (for example: with code debugging tools) through the use of instructions to detect, mislead, and hamper tracing, (d) preventing tampering of said code through the use of scanning to locate alterations, either or both on-disk and in memory either once at the start of execution, or continuously upon certain events, or (e) preventing ID-Data theft

> through the inclusion of secure input/output routines (for example: routines to bypass the standard operating system keyboard calls and use custom-written higher-security routines as a replacement) to replace insecure computer-system routines. Hereafter, the term anti-spy will be used to refer to any combination of one or more of the abovementioned techniques [(a) through (e) or parts thereof] used to prevent eavesdropping.

Drake, Col. 4, lines 47-65.

These portions of Drake cited by the Examiner do not disclose "outputting instructions that obtain the results and provide the results for a user." Recall that claim 1 also states "wherein the results generated indicate whether the observer program is present on the computer system." The portion of in Col. 6 cited by the Examiner is discussing tampering and states "[o]nce tampering is detected, program flow-of-control needs to be changed so that the potential compromise associated with ID-Data theft is avoided. This may be the security-enhanced program terminating with a message indicating that its integrity has been compromised before all of the ID Data is entered." Thus, this section discloses that when tampering has been detected, a message indicating that the program's integrity has been compromised. However, this is not the same as the claim element at issue.

As set forth above, Drake does not disclose every element of claim 1. Claims 2-15 depend directly or indirectly from claim 1. Thus, Appellants respectfully request that the rejection of claims 2-15 be withdrawn for at least the same reasons.

Claims 16

Claim 16 recites "means for accessing observer data," and "observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create a log file from the observing of the observer program." Drake does not disclose this claim element. The Examiner has cited Col. 3, lines 32-67 (quoted above) of Drake as disclosing this claim element. This portion of Drake does not disclose "means for accessing observer data" and "observer data that includes data descriptive of an observer program." It does mention "rogue software eavesdropping" (Col. 3, lines 41-42) and "anti-spy techniques" (Col. 3, lines 43), but these generic terms do not disclose this claim element.

Claim 16 also recites "means for comparing the observer data with memory data to determine whether the observer program is present on the computer system." Drake does not disclose this claim element. The Examiner has cited Col. 6, lines 5-48 of Drake (quoted above) as disclosing this claim element. This portion of Drake does not disclose "means for comparing the observer data with memory data to determine whether the observer program is present on the computer system." This section of Drake discloses aspects of detecting tampering, as the heading in Drake indicates. Recall that claim 16 recited above "observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create data from the observing of the observer program."

Claim 16 further recites "means for outputting the results for a user." Claim 16 also states "wherein the results generated indicate whether the observer program is present on the computer system." Drake does not disclose this claim limitation. The Examiner has cited Col. 6, lines 5-48 (quoted above) and Col. 4, lines 47-65 (quoted above) as disclosing this claim element. These portions of Drake cited by the Examiner do not disclose "means for outputting the results for a user." Recall that claim 16 also states "wherein the results generated indicate whether the observer program is present on the computer system." The portion of in Col. 6 cited by the Examiner is discussing tampering and states "[o]nce tampering is detected, program flow-of-control needs to be changed so that the potential compromise associated with ID-Data theft is avoided. This may be the security-enhanced program terminating with a message indicating that its integrity has been compromised before all of the ID Data is entered." Thus, this section discloses that when tampering has been detected, a message indicating that the program's integrity has been compromised. However, this is not the same as the claim element at issue.

Claims 17-19

Claim 17 recites "accessing observer data, the observer data including data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create a log file from the observing of the observer program." Drake does not disclose this claim element. The Examiner has cited Col. 3, lines 32-

67 (quoted above) of Drake as disclosing this claim element. This portion of Drake does not disclose "accessing observer data, the observer data including data descriptive of an observer program." It does mention "rogue software eavesdropping" (Col. 3, lines 41-42) and "anti-spy techniques" (Col. 3, lines 43), but these generic terms do not disclose this claim element.

Claim 17 also recites "comparing the observer data with memory data read in from memory to determine whether the observer program is present on the computer system." Drake does not disclose this claim element. The Examiner has cited Col. 6, lines 5-48 (quoted above) of Drake as disclosing this claim element. This portion of Drake does not disclose "comparing the observer data with memory data read in from memory to determine whether the observer program is present on the computer system." This section of Drake discloses aspects of detecting tampering, as the heading in Drake indicates. Recall that claim 17 recited above "the observer data including data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create data from the observing of the observer program."

Claim 17 further recites "outputting the results for a user." Claim 17 also states "wherein the results generated indicate whether the observer program is present on the computer system." Drake does not disclose this claim limitation. The Examiner has cited Col. 6, lines 5-48 (quoted above) and Col. 4, lines 47-65 (quoted above) of Drake as disclosing this claim element. These portions of Drake cited by the Examiner do not disclose "outputting the results for a user." Recall that claim 17 also states "wherein the results generated indicate whether the observer program is present on the computer system." The portion of in Col. 6 cited by the Examiner is discussing tampering and states "[o]nce tampering is detected, program flow-of-control needs to be changed so that the potential compromise associated with ID-Data theft is avoided. This may be the security-enhanced program terminating with a message indicating that its integrity has been compromised before all of the ID Data is entered." Thus, this section discloses that when tampering has been detected, a message indicating that the program's integrity has been compromised. However, this is not the same as the claim element at issue.

As set forth above, Drake does not disclose every element of claim 17. Claims 18 and 19 include similar limitations as claim 17. Thus, Appellants respectfully request that the rejections of claims 18 and 19 be withdrawn for at least the same reasons.

Claims 20-32

Claim 20 recites "accessing instructions that access observer data," and "observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system . . . and also operating to create a log file from the observing of the observer program." Drake does not disclose this claim element. The Examiner has cited Col. 3, lines 32-67 of Drake as disclosing this claim element. This portion of Drake does not disclose "accessing instructions that access observer data," and "observer data that includes data descriptive of an observer program." It does mention "rogue software eavesdropping" (Col. 3, lines 41-42) and "anti-spy techniques" (Col. 3, lines 43), but these generic terms do not disclose this claim element.

Claim 20 also recites "altering instructions that alter a file relating to the observer program such that the operation of the observer program is changed." Drake does not disclose this claim element. The Examiner has cited the following portion of Drake as disclosing this claim element:

> Obfuscating inserts can successfully prevent automatic disassembly. Obfuscation is achieved by following unconditional jump instructions (for example, Intel JMP or CLC/JNC combination or CALL (without a return expected) or any flow-of-control altering instruction (which is known not to return to the usual place) with one or more dummy op-code bytes which will cause subsequent op-codes to be erroneously disassembled (for example, the Intel 0xEA prefix will cause disassembly of the subsequent 4 op-codes to be incorrect, displaying them as the offset to the JMP instruction indicated by the 0xEA prefix instead of the instructions they actually represent).
>
> Dummy instructions may also be included to hamper disassembly by deliberately misleading a disassembler into believing a particular flow of control will occur, when in fact it will not.
>
> Flow of control can be designed to occur based upon CPU flag values determined from instructions executed a long time ago. Together with tracing preventing, this makes manual disassembly nearly impossible.

Drake, Col. 5, lines 42-62.

These portions of Drake do not disclose "altering instructions that alter a file relating to the observer program such that the operation of the observer program is changed." These portions of Drake are under the section entitled "Aspect 2. Preventing Disassembly and Examination." Drake, Col. 5, line 36. To help put this section of Drake into context it is helpful to read the paragraph immediately preceding lines 42-62 which reads "[a]s hereinbefore described, it is desirable to hamper disassembly (or de-compilation or reverse engineering) to protect software against eavesdropping and tampering, and to hinder examination of said software which might lead to secret security problems or mistakes being disclosed." Drake, Col. 5, lines 37-41. Thus, Col. 5, lines 42-62 of Drake disclose what to put into a program to prevent disassembly and examination. This is not disclosing any "altering instructions that alter a file relating to the observer program such that the operation of the observer program is changed."

The Examiner has also cited this portion of Drake as disclosing "altering instructions that alter a file relating to the observer program such that the operation of the observer program is changed:"

> Bypassing system routines (eg: in DOS, using direct memory writes instead of DOS system calls to revector interrupts) will further hamper debugging and rogue software monitoring, as will unravelling loop constructs (which will make tracing long and cumbersome). Code checksums and operating-system checks (eg: interrupt table pointers) can be designed to detect debug-breakpoint instruction inserts or other modifications. Using the result of the checksum for some obscure purpose (eg: decryption, or (much later) control-flow changes) will further hamper tracing.

Drake, Col. 8, lines 3-12.

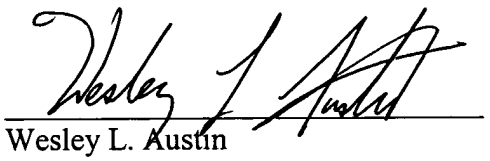
These portions of Drake does not disclose "altering instructions that alter a file relating to the observer program such that the operation of the observer program is changed." Rather, this discloses how to "hamper debugging and rogue software monitoring" and how to "detect debug-breakpoint instruction inserts or other modifications."

As set forth above, Drake does not disclose every element of claim 20. Claims 21-28 depend directly or indirectly from claim 20. Thus, Appellants respectfully request that the rejection of claims 21-28 be withdrawn for at least the same reasons. Claims 29-32 also include

similar limitations as described in relation to Claim 20. Thus, Appellants respectfully request that the rejection of claims 29-32 be withdrawn for at least the same reasons.

Appellants note that claim 1-32 stand or fall together. Therefore, for the reasons discussed above, Appellants assert that the rejection of claims 1-32 is improper. Reversal of the Examiner's rejections and allowance of the pending claims is respectfully requested.

Respectfully submitted,



Wesley L. Austin
Reg. No. 42,273
Attorney for Appellant(s)

Date: January 27, 2006

Wesley L. Austin, Esq.
Trapware Corporation
1244 E. 1650 S.
Bountiful, UT 84010
Telephone: (801) 296-0597

CLAIMS APPENDIX

Listing of Claims involved in the appeal:

1. A system for detecting the presence of an observing program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the system including computer software for running on the computer system, the system comprising:

observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

accessing instructions that access the observer data;

reading instructions that read memory of the computer system to obtain memory data;

comparing instructions that compare the observer data with memory data read in from memory to determine whether the observer program is present on the computer system;

generating instructions that generate results from the comparing, wherein the results generated indicate whether the observer program is present on the computer system; and

outputting instructions that obtain the results and provide the results for a user.

2. The system of claim 1 wherein the reading instructions read the memory of the computer system by querying the operating system of the computer system for the tasks running and by examining task information provided by the operating system.

3. The system of claim 1 wherein the outputting instructions provide the results to a user through a graphical user interface.

4. The system of claim 1 wherein the reading instructions read the memory of the computer system by querying the file system of the computer system for the files located on storage media and by examining file information provided by the file system.

5. The system of claim 1 wherein the reading instructions read the memory of the computer system by opening a file located on storage media and by examining contents of the file.

6. The system of claim 1 wherein the observer data includes data descriptive of a plurality of observer programs and wherein the system compares the observer data with the memory data to determine whether any known observer program is present.

7. The system of claim 1 further comprising countermeasure instructions wherein the countermeasure instructions alter the operation of the observer program.

8. The system of claim 7 wherein the countermeasure instructions alter the operation of the observer program by observer program configuration data.

9. The system of claim 7 wherein the countermeasure instructions alter the operation of the observer program by altering a file on the computer system.

10. The system of claim 7 wherein the countermeasure instructions alter the operation of the observer program by altering reporting data generated by the observer program.

11. The system of claim 7 wherein the countermeasure instructions alter the operation of the observer program by replacing reporting data generated by the observer program.

12. The system of claim 7 wherein the countermeasure instructions alter the operation of the observer program by replacing a file of the observer program.

13. The system of claim 1 wherein the observer data includes data descriptive of observing activity typical of observing programs and wherein the system compares the observer data with the memory data to determine whether any known observer program is present.

14. The system of claim 1 further comprising the observer data, wherein the observer data includes a list of files and modules that are part of the observer program software, and wherein the reading instructions read the memory of the computer system by querying

the operating system of the computer system for the tasks running and by examining task information provided by the operating system, and wherein the reading instructions also read the memory of the computer system by querying the file system of the computer system for the files located on storage media and by examining file information provided by the file system, and wherein the outputting instructions provide the results to a user through a graphical user interface.

15. The system of claim 1 wherein the system is made available over a computer network through a web site.

16. A system for detecting the presence of an observing program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the system including computer software for running on the computer system, the system comprising:

observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

means for accessing the observer data;

means for reading memory of the computer system to obtain memory data;

means for comparing the observer data with memory data to determine whether the observer program is present on the computer system;

means for generating results from the comparison, wherein the results generated indicate whether the observer program is present on the computer system; and

means for outputting the results for a user.

17. A method for detecting the presence of an observing program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the method being implemented through computer software for running on the computer system, the method comprising the steps of:

accessing observer data, the observer data including data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

reading memory of the computer system to obtain memory data;

comparing the observer data with memory data read in from memory to determine whether the observer program is present on the computer system;

generating results from the reading and comparing, wherein the results generated indicate whether the observer program is present on the computer system; and

outputting the results for a user.

18. A computer-readable medium containing instructions for detecting the presence of an observing program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, wherein the instructions comprise executable instructions for implementing a method comprised of the steps of:

accessing observer data, the observer data including data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

reading memory of the computer system to obtain memory data;

comparing the observer data with memory data read in from memory to determine whether the observer program is present on the computer system;

generating results from the reading and comparing, wherein the results generated indicate whether the observer program is present on the computer system; and

outputting the results for a user.

19. The computer-readable medium of claim 18 wherein the computer-readable medium is a data transmission medium.

20. A system for altering the operation of an observer program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the system including computer software for running on the computer system, the system comprising:

observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

accessing instructions that access the observer data;

reading instructions that read memory of the computer system to obtain files relating to the observer program;

altering instructions that alter a file relating to the observer program such that the operation of the observer program is changed.

21. The system of claim 20 further comprising an observer detection program.

22. The system of claim 20 further comprising inputting instructions that display to a user options regarding the altering and that take input from the user relating to the options.

23. The system of claim 20 wherein the altering instructions alter the operation of the observer program by altering observer program configuration data.

24. The system of claim 20 wherein the altering instructions alter the operation of the observer program by altering a file on the computer system.

25. The system of claim 20 wherein the altering instructions alter the operation of the observer program by altering reporting data generated by the observer program.

26. The system of claim 20 wherein the altering instructions alter the operation of the observer program by replacing reporting data generated by the observer program.

27. The system of claim 20 wherein the altering instructions alter the operation of the observer program by replacing a file of the observer program.

28. The system of claim 20 wherein the system is made available over a computer network through a web site.

29. A system for altering the operation of an observer program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing

on the computer system, the system including computer software for running on the computer system, the system comprising:

observer data that includes data descriptive of an observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

means for accessing the observer data;

means for reading memory of the computer system to obtain files relating to the observer program; and

means for altering a file relating to the observer program such that the operation of the observer program is changed.

30. A method for altering the operation of an observer program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, the method being implemented through computer software for running on the computer system, the method comprising the steps of:

accessing observer data, the observer data including data descriptive of the observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

reading memory of the computer system to obtain files relating to the observer program; and

altering a file relating to the observer program such that the operation of the observer program is changed.

31. A computer-readable medium containing instructions for altering the operation of an observer program on a computer system, wherein the observing program is programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and to create data from the observing on the computer system, wherein the instructions comprise executable instructions for implementing a method comprised of the steps of:

accessing observer data, the observer data including data descriptive of the observer program, the observer program being programmed to observe a user's activities on the computer system by monitoring user input entered through a user input device and also operating to create a log file from the observing of the observer program;

reading memory of the computer system to obtain files relating to the observer program; and

altering a file relating to the observer program such that the operation of the observer program is changed.

32. The computer-readable medium of claim 31 wherein the computer-readable medium is a data transmission medium.

EVIDENCE APPENDIX

NONE.

RELATED PROCEEDINGS APPENDIX

NONE.